

# Visual Room-Awareness for Humanoid Robot Self-Localization

Markus Bader      Johann Prankl  
Markus Vincze

Automation and Control Institute, Vienna University of Technology, Austria

**Abstract.** Humanoid robots without internal sensors such as a compass tend to lose their orientation after a fall. Furthermore, re-initialisation is often ambiguous due to symmetric man-made environments. The room-awareness module proposed here is inspired by the results of psychological experiments and improves existing self-localization strategies by mapping and matching the visual background with colour histograms. The matching algorithm uses a particle-filter to generate hypotheses of the viewing directions independent of the self-localization algorithm and generates confidence values for various possible poses. The robot's behaviour controller uses those confidence values to control self-localization algorithm to converge to the most likely pose and prevents the algorithm from getting stuck in local minima. Experiments with a symmetric Standard Platform League RoboCup playing field with a simulated and a real humanoid NAO robot show the significant improvement of the system.

## 1 Introduction

Rule changes in 2012 in the RoboCup *Standard Platform League* (SPL) resulted in a symmetrical playing field with identical goals. Therefore, participating teams must keep track of the direction of play during the game to prevent own goals. This poses a problem because sensors like an electronic compass are not allowed. Figure 1 shows the humanoid robot used for the experiments. Here, the approach presented in order to confront this challenge is inspired by psychological beliefs and recent experimental results with symmetric environments [7], [8] and [10] which proved the existence of spontaneous reorientation in animals and humans. These experiments showed that the geometric structure of a room has a strong influence on reorientation capabilities. Lee et al. [10] proved that the geometric impression of a room can be altered by using 2D shapes (dots of two sizes) printed on walls, supporting or suppressing the subjective geometric impression. Three-year-old children spontaneously re-oriented in these experiments to the correct corner when the larger-sized dots were on the longer wall of a rectangular room, emphasising the 3D impression. The influence of the geometric impression of a space on the ability to reorient, as proven in the aforementioned psychological experiments, was integrated into the approach here through the mapping of the visual

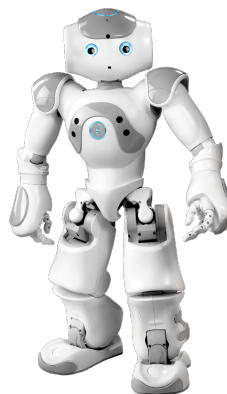


Figure 1: Aldebaran NAO v4, the robot has no electronic compass.

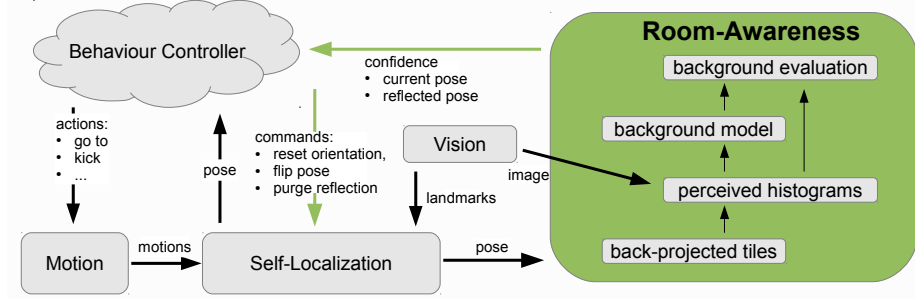


Figure 2: The room-awareness module, its integration, in- and output and internal work-flow.

background of the robot’s surroundings beyond the known playing field. This is done by using colour histograms in order to keep track of the robot’s orientation in the room-awareness module, which estimates the robot’s orientation, as a separate module. This enables the computation of independent confidence values for the current pose generated by the self-localization module. Fig. 2 shows the integration and the separation of the room-awareness and self-localization module. The robot’s Behaviour Controller (BC) uses these confidence values to supervise the self-localization module with three simple commands:

- **flip pose**  
This command changes the robot’s belief of being in a symmetric reflection.
- **purge reflection**  
This command triggers the self-localization algorithm to remove pose beliefs in symmetric reflections.
- **reset orientation**  
This command resets the self-localization algorithm’s belief in the robot’s orientation. The robot’s position belief remains untouched. It is used after the internal sensors detect a fall.

Therefore, the BC is able to optimise the particle filter used by the self-localization algorithm [9], [12]. A wrong particle cluster, for example, is removed using the *purge reflection* signal if the confidence of the current pose wins over the reflected pose confidence (as in Fig. 6(b)), or a *flip pose* will be triggered if the reflected pose wins. The scientific contributions of this paper are a room-awareness module which mimics a human-like belief of one’s orientation and its integration, allowing the robot’s BC to trigger a spontaneous reorientation.

The next section presents related work on the topic followed by a detailed description of the approach and supporting results in Section 4. The results are summarised in Section 5.

## 2 Related Work

The strategies presented by different RoboCup-Teams during the past year can be classified into two main groups.

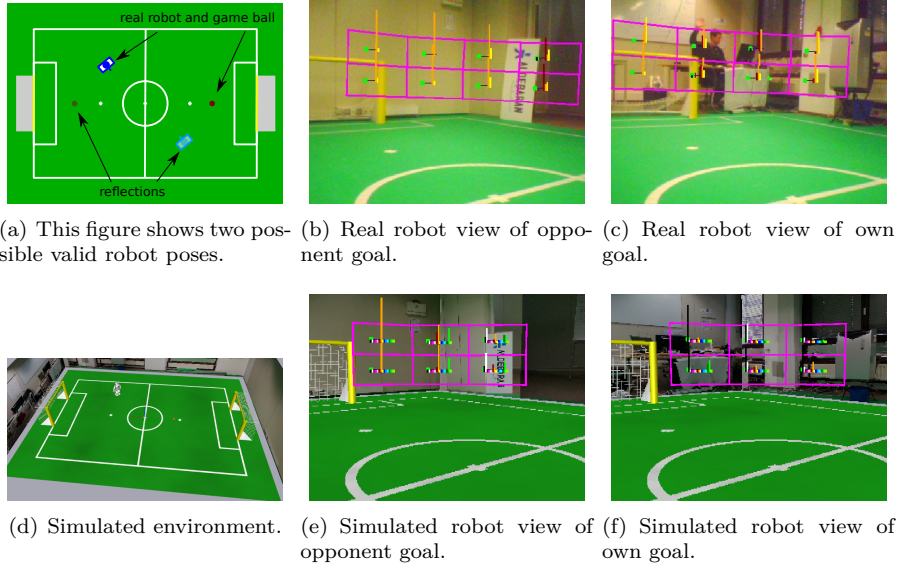


Figure 3: These figures present the scene perception with a real and simulated robot. The tiles projected depend on the robot's locations in (c).

- Strategies using **non-static features on the playing field**  
Recognized objects like the game ball or other robots, which are needed to play soccer, are used as non-static features in order to overcome the symmetry problem.
- Strategies using **features beyond the playing field**  
New features in the background are used in addition to features on the symmetric playing field.

## 2.1 Non-Static Features on the Playing Field

The *non-static features strategy* uses objects like the game ball or other detected objects in addition to features on the symmetric playing field. For example Fig. 3(d) shows a single robot and a game ball on the playing field. Fig. 3(a) show the two possible poses. The robot can determine a correct pose by knowing the ball position. This idea can be used with a single robot by tracking all objects of interest, but it works optimally with team communication, because perceived ball and robot locations are shared. This approach is therefore similar to multi-robot localization [6]. The system starts to fail if the team has only one player left and this robot falls, because the stability of the non-static features cannot be guaranteed after a certain period of time. Some teams (e.g. the Dortmund Devils) have already integrated non-static features into their Kalman-Filter based localization [5].

## 2.2 Features Beyond the Playing Field

Similar to humans, this strategy uses features beyond the symmetric playing field. The idea is to identify outstanding features in the background and map them using *Self Localization and Mapping* (SLAM) approaches. Such techniques

are widely used in robot localization. For instance, Anati [1] trained an object-detection algorithm to recognise, among others, clocks and trash cans in order to determine a robot's location within a train station. State of the art object detection algorithms are often based on interest points, i.e., salient image regions and a descriptor. For example the most popular interest points are Lowes' SIFT [11] and Bays' SURF [4]. Because of the limited computational power in RoboCup SPL, it is impossible to use such object detection algorithm during a soccer game. As an alternative, Anderson [2] presented a simplified 1D SURF descriptor to map backgrounds. In [3] Bader et al. use colour histograms as descriptors. However, they did not propose a reliable strategy for matching those descriptors over time.

The challenges for any strategy which uses the background for localization can be summarized by:

- **Training:** When should the background be trained?
- **Matching:** When should it be used for matching?

This work proposes the use of colour histogram based image features and a strategy for finding a balance between learning and training those features. It might be said that other features, e.g image gradient information, could also be used for mapping, but for the purposes of this study, only colour histograms were implemented.

### 3 Room-Awareness

The technique proposed here improves existing particle filter based self-localization algorithms by allowing the robot's BC to change internal localization hypotheses. Fig. 2 shows the integration of the room-awareness module and the control channels to an the self-localization algorithm used (green).

The external placement of the room-awareness module and the BC as the decision making unit allow for further integration of other techniques, e.g. non-static features like the game ball, without changing a well established localization technique.

The room-awareness approach requires initial knowledge, such as the robot's pose or a previously learned visual background model. The following sections describe the sub-modules needed in order to realise the approach.

#### 3.1 Perceived Colour Histograms and Background Tiles

The *perceived colour histograms* are linked to tiles on a virtual surrounding, see Fig 4. The cylinder-shaped virtual wall is composed of multiple rows and columns of rectangular tiles. Figure 5 shows an example of a colour histogram used to augment the robot's views in all other figures. Colour bin values are drawn upwards and variance of each bin is drawn downwards. Due to symmetry, the back-projected tiles appear equal for all reflected poses on the camera image, see Fig. 3(b,c,e,f). The color histograms linked to the *background tiles* can be used to disambiguate the viewing direction. Perceived histograms are drawn at the bottom left, outside the background model in Fig. 6(a), without variance.

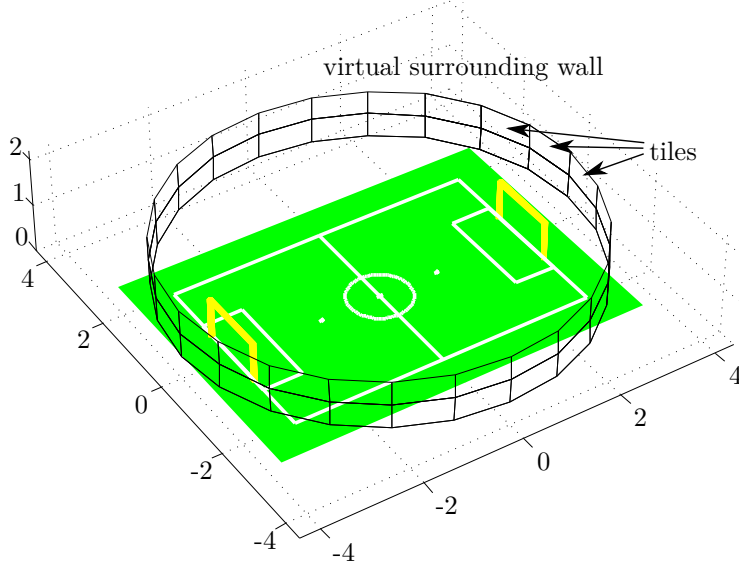


Figure 4: Surrounding wall of tiles in multiple rows and columns in the shape of a cylinder.

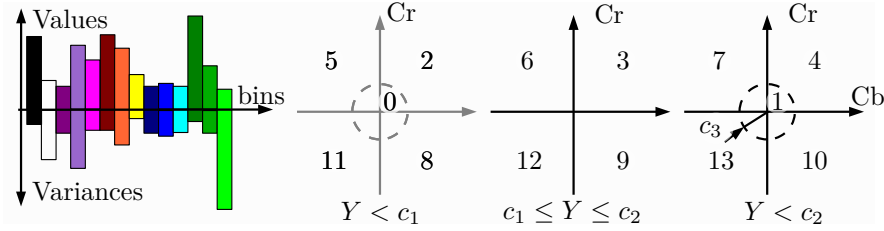


Figure 5: A color histogram and bin values are drawn upwards, and the related variances downwards. Such histograms are used for modelling the background. The bins are defined by the axis sign and three thresholds  $c_1$ ,  $c_2$  and  $c_3$ , as drawn on the three YCrCb colour space slices.

### 3.2 The Background Model

The *background model* is trained online with perceived histograms by using a moving average update strategy. To stabilize the model, tiles which are blocked by other robots or observed from a certain viewing angle are ignored. The moving average update strategy allows for the computation of variance for each colour histogram bin in order to detect unstable areas. The equations for computing the moving average  $\mu$  and variance  $\sigma$  are shown in Eq. (1) and (2), but if a tile is seen for the first time, the perception is copied. Increasing  $N$  leads to a more stable model but to a lower rate of adaptation to environmental changes.

$$\mu_{new} = \frac{N\mu_{last} + x_{mess}}{N + 1} \quad (1)$$

$$\sigma_{new}^2 = \frac{1}{N + 1} \left( N\sigma_{last}^2 + \frac{N}{N + 1} * (\mu_{last} - x_{mess})^2 \right) \quad (2)$$

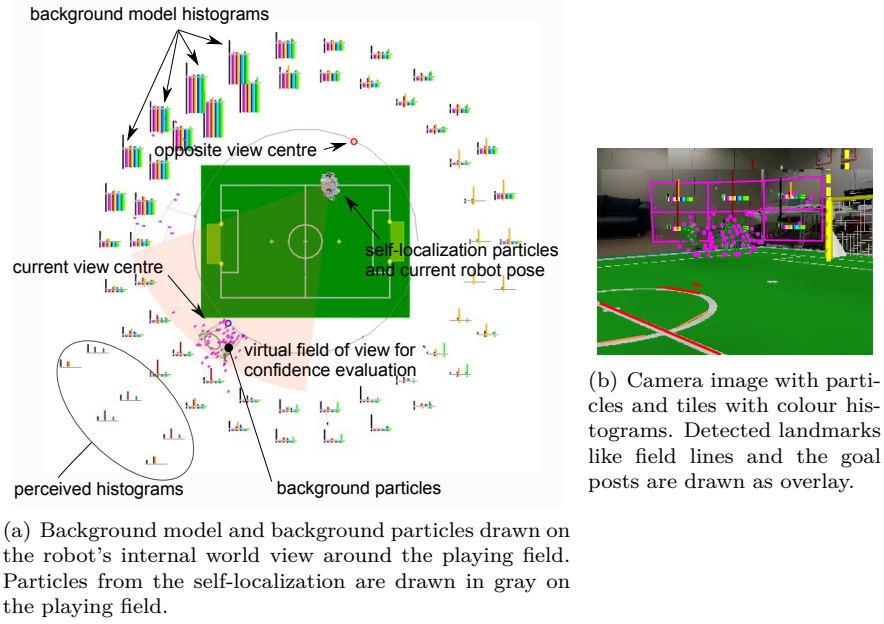


Figure 6: The robot's internal world view and the related camera image.

The training can be interrupted by the robot's BC to avoid learning from incorrect models, as when the robot falls or during a penalty in the soccer game. A trained background model with colour histograms around the playing field is shown in Fig. 6(a). The two circles of histograms indicate the two rows of tiles cylindrically arranged around the playing field.

### 3.3 The Background Evaluation

This sub-module uses the robot pose information and the perceived colour histograms to estimate the current viewing direction based on the background model. The estimation is done by using a particle-filter where each particle describes a viewpoint hypothesis on the cylindrically modelled wall. The particle weights are computed by comparing the colour histograms of the perceived tiles with model histograms of viewpoints estimated by a specific particle. All particles are updated by linearising the current rotational motion at the current view center plus some additional white noise to compensate for model discrepancy. New particles are injected at the current best matching position to prevent the filter from getting stuck in local minima. Fig. 6(a) indicates particles with magenta dots around the playing field and the detected cluster center is drawn as a gray ellipse. The same particles are visualized in the current perceived image, Fig. 6(b) (black).

### 3.4 Confidence Values

Multiple confidence values are computed by counting the number of background particles within certain angle ranges. Confidence for the current pose is computed by counting the number of particles within a virtual field of view around the current view center, shown in Fig. 6(a). The opposite pose confidence is

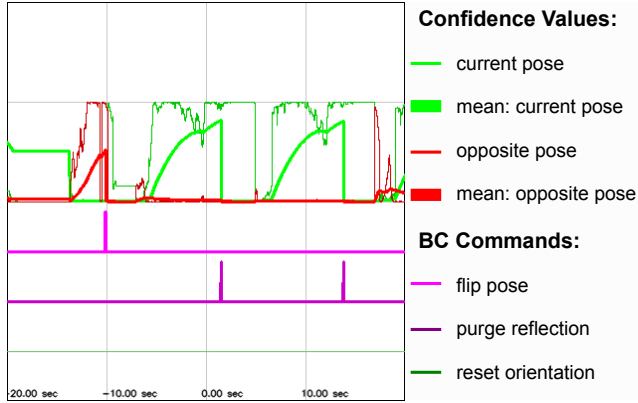


Figure 7: History of past confidence values and BC's command signals. The BC recently triggered a *flip* because the robot was placed incorrectly. This was followed by *purge reflection* because the awareness module indicated a high current pose confidence over the reflected pose confidence.

computed similarly but with the opposite view center. A moving average algorithm uses the previous values to suppress incorrect estimates and to assure a smoother confidence value. Fig. 7 shows the previous measurements as signals over time, together with the BC command signals.

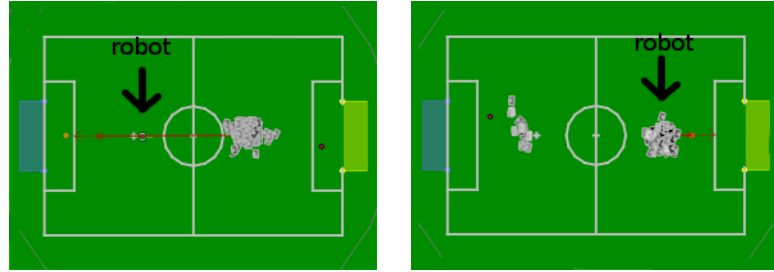
## 4 Results

Two test scenarios were used on a simulated and real robot to measure this approach's improvement on a system without a room-awareness module. For both tests the robot was placed on the default soccer start position next to the playing field, but the particle-filter based self-localization was initialized with the opposite "incorrect" pose, and sides were switched after half the trials. A system which uses only the symmetric playing field for localization would converge only by accident to the correct pose and would normally fail. In the *first test* the robot was allowed to move only its head. In the *second test* the robot had to walk from one penalty position to the other and vice versa, based on its own localisation. During all of the trials the time it took for the BC to trigger a control command to optimize or correct the self-localization's particle distribution was measured. Table 1 shows the number of trials and the average time it took for the BC to create a signal. One can see that the system tends to

Simulated Robot	trials	flip	purge	failed
not moving	20 trials	70%	15%	15%
average time to a signal	18.1 sec	12.9 sec	57.5 sec	> 200 sec
moving	10 trials	80%	20%	0%
average time to a signal	20.2 sec	17.1 sec	34.5 sec	-
Real Robot	trials	flip	purge	failed
not moving	20 trials	90%	0%	10%
average time to a signal	23.1 sec	23.1 sec	-	> 200 sec
moving	10 trials	100%	0%	0%
average time to a signal	33.5 sec	33.5	-	-

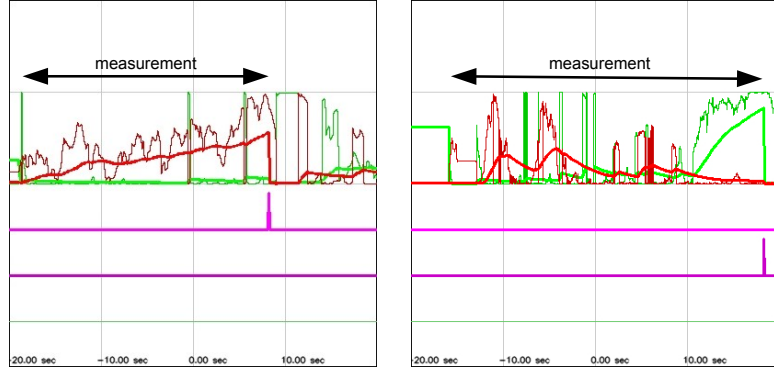
Table 1: Results from a simulated and real robot

fail up to 20% if the robot is not moving. This happens because the background is trained online and the wrong background is assumed as correct over a certain time. Two ways in which the BC corrects the self-localization distribution were also observable, Fig. 8. If the particles are on the wrong pose, the system triggers a flip. A purge was called for if the filter accidentally formed a correct growing particle cluster. The reason why the system on the real robot never triggered a purge in these tests is because, for both the simulation and the real robot, the room-awareness parameters were fixed to generate comparable results. On the real robot smoother confidence values were experienced than on the simulated, visible in Fig. 8(c), for which the noisier and smoother image module was held responsible. Overall, it was shown that the proposed room-awareness module improves an existing self-localization algorithm.



(a) All of the self-localization particles are on the incorrect pose.

(b) Most of the self-localization particles are on the correct pose.



(c) A *flip* solved the problem of an incorrect pose. Data was recorded on a real robot.

(d) A *purge* optimised the distribution. Data was recorded on a simulated robot.

Figure 8: Two cases were observed in which the system optimised the particle distribution after an incorrect initialization on a simulated robot.

## 5 Conclusion

In this paper, a room-awareness module inspired by the results of psychological studies was presented which mimics a human belief of current pose and triggers a so-called spontaneous reorientation. Results of tests with a humanoid robot on a RoboCup SPL playing field supported the initial belief in recognition of an incorrect estimated pose due to symmetric environments by mapping the surrounding environment with colour histograms. In addition, a less-cluttered



particle distribution in the self-localization used was generated during normal operation by allowing the BC to interfere with the self-localization by selectively moving or removing particle clusters caused by a symmetric environment. However, multiple issues remain open, e.g. an optimal count of color histograms, a different feature for modelling environments such as gradient information, the virtual wall shape used, and the implementation of an optimal search pattern to force the robot to look at areas with the most distinctive background.

## Acknowledgments

The research leading to these results has received funding from the Austrian Science Foundation under grant agreement No. 835735 (TransitBuddy)

## References

- [1] Roy Anati, Davide Scaramuzza, Konstantinos Derpanis, and Kostas Daniilidis. Robot localization using soft object detection. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 4992–4999, May 2012.
- [2] Peter Anderson, Yongki Yusmanthia, Bernhard Hengst, and Arcot Sowmya. Robot localisation using natural landmarks. In *RoboCup 2012: Robot Soccer World Cup XVI Preproceedings*. RoboCup Federation, 2012.
- [3] Markus Bader, Helmut Brunner, Thomas Hamböck, Alexander Hofmann, and Markus Vincze. Colour Histograms as Background Description: An approach to overcoming the Uniform-Goal Problem within the SPL for the RoboCup WC 2012. In *Proceedings of the Austrian Robotics Workshop (ARW-12)*, 2012.
- [4] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In Aleš Leonardis, Horst Bischof, and Axel Pinz, editors, *Computer Vision – ECCV 2006*, volume 3951 of *Lecture Notes in Computer Science*, pages 404–417. Springer Berlin / Heidelberg, 2006.
- [5] Stefan Czarnetzki and Carsten Rohde. Handling heterogeneous information sources for multi-robot sensor fusion. pages 133 –138, sep. 2010.
- [6] Dieter Fox, Wolfram Burgard, Hannes Kruppa, and Sebastian Thrun. A probabilistic approach to collaborative multi-robot localization. *Autonomous Robots*, 8(3):325–344, 2000.
- [7] L. Hermer and E. S Spelke. A geometric process for spatial reorientation in young children. *Nature*, 370:57–59, 1994.
- [8] Janellen Huttenlocher and Stella F. Lourenco. Coding location in enclosed spaces: is geometry the principle? *Developmental Science*, 10(6):741–746, 2007.
- [9] Tim Laue and Thomas Röfer. Particle filter-based state estimation in a competitive and uncertain environment. In *Proceedings of the 6th International Workshop on Embedded Systems*. Vaasa, Finland, 2007.

- [10] Sang Ah Lee, Nathan Winkler-Rhoades, and Elizabeth S. Spelke. Spontaneous reorientation is guided by perceived surface distance, not by image matching or comparison. *PLoS ONE*, 7:e51373, 12 2012.
- [11] Gareth Loy and Jan-Olof Eklundh. Detecting symmetry and symmetric constellations of features. *Lecture Notes in Computer Science*, 3952/2006:0302–9743 (Print) 1611–3349 (Online), July 2006.
- [12] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.